

CLAIMS

What is claimed is:

1. In a computer system, a method for generating an adapter/stub, comprising:
 - identifying a machine state input parameter for a machine state;
 - identifying a call to compiled code input parameter for a call to compiled code;
 - mapping the machine state input parameter and the machine state to the call to compiled code input parameter; and
 - mapping the machine state and a return value to an exit point of an interpreter to compiled code adapter.
2. A method as recited in claim 1, further comprising:
 - providing a stub representation to a compiler for compilation; and
 - generating object code based upon the compilation.
3. A computer program product that implements an apparatus for generating an adapter/stub, comprising:
 - computer code that identifies a machine state input parameter for a machine state;
 - computer code that identifies a call to compiled code input parameter for a call to compiled code;
 - computer code that maps the machine state input parameter and the machine state to the call to compiled code input parameter; and

computer code that maps the machine state and a return value to an exit point of an interpreter to compiled code adapter; and
a computer readable medium that stores the computer codes.

4. A computer program product as recited in claim 3, further comprising:
computer code that provides a stub representation to a compiler for compilation; and
computer code that generates object code based upon the compilation.
5. The computer program product of claim 4, wherein the computer readable medium is selected from the group consisting of CD-ROM, floppy disk, tape, flash memory, system memory, hard drive, and data signal embodied in a carrier wave.
6. In a computer system, an apparatus for processing a bytecode in a runtime environment, comprising:
means for receiving the bytecode;
means for determining if the bytecode is to be interpreted or executed;
if the bytecode is to be executed,
means for determining if an adapter is required in order to process the bytecode;
means for determining if the adapter is located in an adapter library if the adapter is required;
means for generating the adapter if it is determined that the required adapter is not in the library;
means for storing the generated adapter in the library;

means for providing the adapter to the runtime environment;

means for executing the bytecode;

if the bytecode is to be interpreted,

means for determining if the adapter is required; and

means for interpreting the bytecode when the adapter is not required; and

means for generating the adapter when the adapter is required.

7. A computer program product that processes a bytecode in a runtime environment, comprising:

computer code for receiving the bytecode;

computer code for determining if the bytecode is to be interpreted or executed;

if the bytecode is to be executed,

computer code for determining if an adapter is required in order to process the bytecode;

computer code for determining if the adapter is located in an adapter library

if the adapter is required;

computer code for generating the adapter if it is determined that the required adapter is not in the library;

computer code for storing the generated adapter in the library;

computer code for providing the adapter to the runtime environment;

computer code for executing the bytecode;

if the bytecode is to be interpreted,

computer code for determining if the adapter is required; and

computer code for interpreting the bytecode when the adapter is not required;
computer code for generating the adapter when the adapter is required; and
a computer readable medium that stores the computer codes.

8. The computer program product of claim 7, wherein the computer readable medium is selected from the group consisting of CD-ROM, floppy disk, tape, flash memory, system memory, hard drive, and data signal embodied in a carrier wave.

卷之三